CoD: A Diffusion Foundation Model for Image Compression

Supplementary Material

This document provides the supplementary material to **Co**mpression-oriented **D**iffusion foundation model, **CoD**. Beyond additional training details and extended results, we further investigate its characteristics and explore additional downstream codec applications.

The key conclusions are summarized below:

- **Extreme 64-bit compression**: CoD compresses image into only 64 bits while preserving correct semantics.
- **One-step diffusion codec**: Finetuning one-step CoD achieves real-time and SOTA performance, comparable to StableCodec [25] and OneDC [22].
- **CoD as perceptual loss**: CoD-based perceptual loss significantly improves MS-ILLM [11].
- **$\mathcal{X}$-prediction advantage**: Using $\mathcal{X}$-prediction yields better performance than $\mathcal{V}$-prediction for CoD (pixel).

A. Towards 64-Bit Image Compression

In this paper, we primarily discuss CoD at 0.0039 bpp, which corresponds to 1024 bits for a 512×512 image. Results demonstrate that the shape, color, and high-level semantic fidelity are well preserved under this bitrate. In this section, we further explore a more extreme compression of 64 bits for a 512×512 image (i.e., 0.00024 bpp) to analyze the boundary of semantic-level compression. To achieve 64 bits, our encoder downsamples the original image to $1/128$ of its original resolution (4×4 patches) and uses a codebook size of $2^4 = 16$. Since latent-space CoD performs better at lower bitrates (Figure 5 in the paper), we train our 64-bit CoD on latent space.

Figure 1. Evaluation of 64-bit CoD on Kodak at 512×512.

We evaluate our 64-bit CoD on the Kodak dataset in Figure 1. Surprisingly, under such an extremely low bitrate, CoD successfully reconstructs the correct semantics of the original image. By contrast, Stable Diffusion based codecs DDCM [12] and DiffC [18] fail to reconstruct correct semantics even at a fourfold ($4\times$) higher bit cost. We further leverage DiffC to evaluate the downstream compression performance on Kodak [1] in Figure 2, where CoD-based DiffC significantly outperforms other Stable Diffusion based codecs. Our scheme achieves a FID of 70 with only less than 10% of the bits of prior codecs, highlighting the extreme compression capability of 64-bit CoD.

B. Towards One-Step Diffusion Compression

B.1. Preliminary

Recently, one-step diffusion models have demonstrated effectiveness in generative image compression. Unlike multi-step diffusion, which optimizes each diffusion timestep separately, one-step diffusion can be trained end-to-end, enabling better overall performance. OSCAR [4] fine-tuned a multi-step Stable Diffusion to function as a one-step image codec. StableCodec [25] leverages a one-step SD-Turbo as the foundation model, achieving significantly improved performance. Similarly, OneDC [22] employs a one-step DMD-distilled [23] Stable Diffusion as the foundation model for one-step diffusion compression. The success of these approaches motivates our exploration of CoD in the context of one-step diffusion foundation models.

Since the SD-Turbo distillation process is not fully public, we follow DMD [23] to distill our diffusion model. Given a one-step diffusion network $G_\theta(z)$ that generates x using random noise z, DMD proves that the KL divergence between the real and fake distributions can be expressed as a distribution-matching loss:

$$\nabla_\theta D_{KL} = \mathbb{E}_{\substack{z\sim\mathcal{N}(0;\mathbf{I}) \\ x=G_\theta(z)}} \left[-\left(s_{\text{real}}(x) - s_{\text{fake}}(x)\right) \frac{dG}{d\theta} \right], \quad (1)$$

where $s_{\text{real}}(x)$ and $s_{\text{fake}}(x)$ are the score functions [16] of their respective distributions. By adding random noise to x, the score can be predicted by a diffusion model. DMD leverages a pretrained multi-step diffusion model for the real score and dynamically trains a diffusion model to estimate the fake score for one-step samples x.

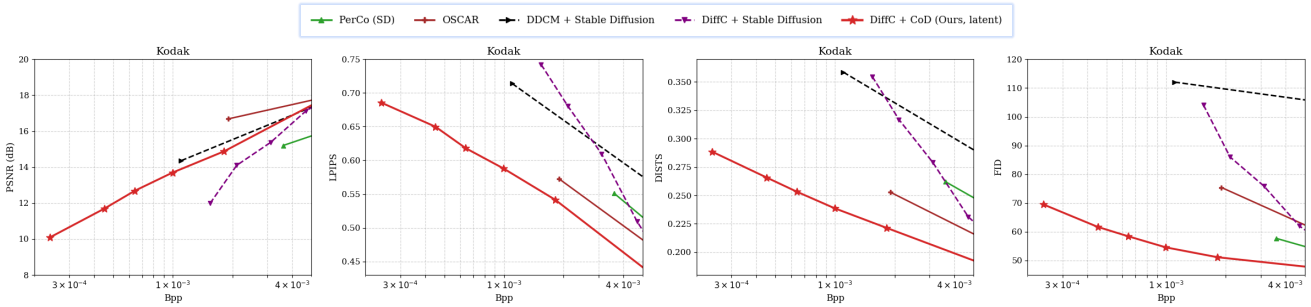

Figure 2. Rate-Distortion curves for 64-bit CoD and Stable Diffusion based codecs on Kodak at 512×512.

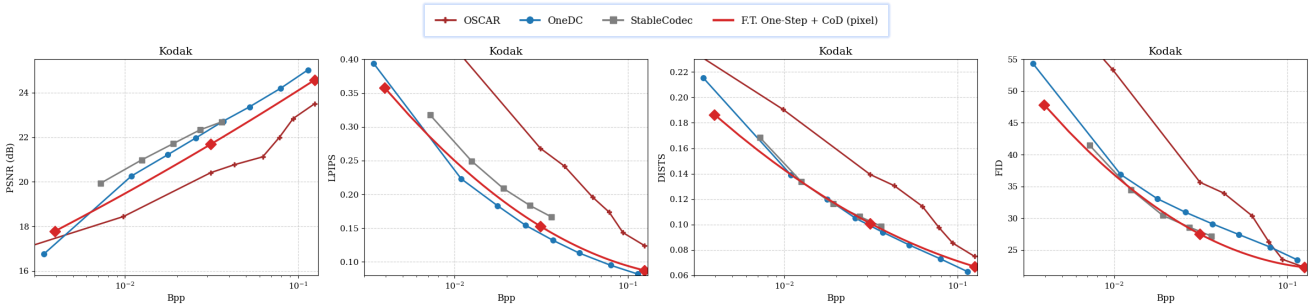

Figure 3. Rate-Distortion curves for finetuned one-step CoD and other one-step diffusion codecs on Kodak at 512×512.

Figure 4. Evaluation of one-step CoD on Kodak at 512×512.

B.2. Distilling a CoD

We train a one-step CoD using a combination of losses and multi-step CoD-based DMD loss:

$$\mathcal{L}_{\text{OS}} = \mathcal{L}_{\text{pixel}} + \mathcal{L}_{\text{perc}} + \lambda \cdot \mathcal{L}_{\text{REPA}} + \beta \cdot \mathcal{L}_{\text{C}}, \quad (2)$$

where $L_{\text{pixel}} = \mathcal{L}_1 + \mathcal{L}_{\text{LPIPS}}$ consists of pixel-space L1 loss and the LPIPS [24] loss, $L_{\text{perc}} = 2 \cdot \mathcal{L}_{\text{DMD}} + 0.01 \cdot \mathcal{L}_{\text{GAN}}$ consists of a DMD loss and a patch GAN loss [2], $\mathcal{L}_{\text{REPA}}$ is the representation alignment loss and $\mathcal{L}_{\text{C}}$ is th codebook commitment loss. We set $\lambda = 0.5$ and $\beta = 0.25$.

For simplicity, we optimize only the pixel-space CoD, allowing pixel-space losses to be computed directly. Following the DMD procedure, we update the one-step CoD every 10 steps, while the remaining 9 steps train the fake score diffusion model. The model is trained for 100K steps (i.e., 10K steps for the one-step CoD) with a batch size of 32, which takes approximately 96 A100 GPU hours in total, which is much less than 1664 A100 GPU hours of DMD-based Stable Diffusion distillation. The learning rates for one-step CoD and fake score network are set to 10^{-5}.

B.3. Finetuning to Higher Bitrates

In the previous section, we distilled the one-step CoD model at 0.0039 bpp. We further observe that the distilled model can be efficiently adapted to higher bitrates. Following PerCo and OSCAR, we adjust both the downsampling scale in the condition encoder and the codebook size according to the target bitrate. Specifically, we train two variants: 0.0312 bpp with $16\times$ downsampling and codebook size 256, and 0.125 bpp with $8\times$ downsampling and codebook size 256.

Stage I: initialization. We randomly set the parameters of the condition encoder, codebook, and condition decoder, while loading the pre-trained multi-step CoD weights for the diffusion module. During this warm-up phase, the diffusion module is trained using LoRA [5] with rank 32. To accelerate convergence, we remove the L_{perc} term from L_{OS} and train for 200K steps with a learning rate of 10^{-4}.

Stage II: Fine-tuning. We then fine-tune all model components jointly using the full L_{OS} for another 100K steps with a batch size of 32 and a learning rate of 10^{-5}. With perceptual supervision restored, the fine-tuned one-step CoD achieves significantly improved performance at higher bitrates.

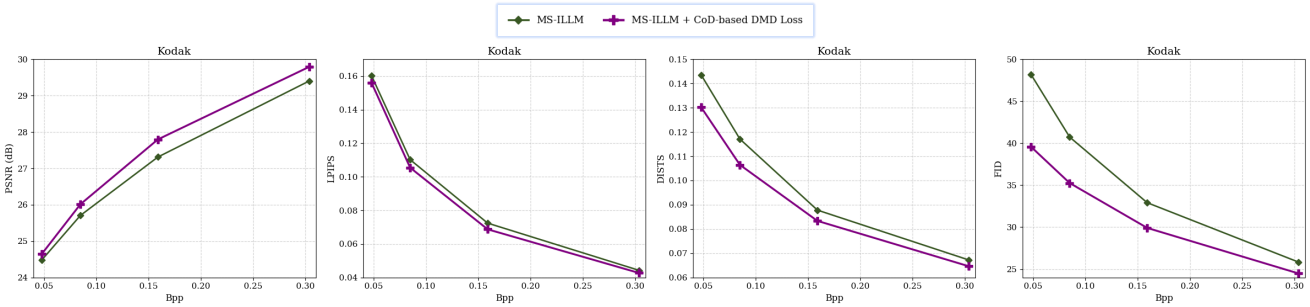

Figure 5. Rate-Distortion curves for MS-ILLM [11] and finetuning MS-ILLM decoder with CoD-based DMD loss on Kodak at 512×512.

B.4. Evaluation

Comparison with one-step codecs. We evaluate the finetuned one-step CoD on the Kodak dataset (Figure 3). It yields competitive reconstruction quality compared with SOTA one-step diffusion codecs such as OneDC and StableCodec, particularly at ultra-low bitrates. Notably, our method does not rely on any pretrained components from Stable Diffusion (e.g., one-step models or SD-VAEs) to reach this performance. We currently adopt fixed-length coding for the codebook indices for simplicity. Further gains can be achieved by introducing an entropy model or applying latent compression [6]. Visual comparisons in Figure 4 further show that one-step CoD preserves higher fidelity than both multi-step CoD and OneDC.

Complexity Analysis. DiT injects conditios and timesteps through AdaLN-Zero layers, which account for more than 200M parameters in CoD. However, CoD concatenates compression conditions directly with the noised input along the channel dimension instead of injecting them through AdaLN-Zero (see Section **??**). Under the one-step setting, the timestep is fixed to 0, making all AdaLN-Zero outputs constant and therefore redundant. Thus, we precompute these constants and remove the AdaLN-Zero modules to further reduce computation and memory access. As shown in Table 4, one-step CoD processes a 512×512 image in only 25.2 ms, which is even faster than a single step of Stable Diffusion used in OneDC and StableCodec. This demonstrates its potential in real-time application.

Methods	PSNR	LPIPS	DISTS	FID
MS-ILLM	**21.43 dB**	0.403	0.271	92.5
+ CoD-based DMD loss	21.08 dB	**0.376**	**0.248**	**80.92**

Table 1. Finetuning the decoder of MS-ILLM using CoD as a perceptual supervision. Evaluated on Kodak at 512×512. BPP=0.011.

Figure 6. Finetuning MS-ILLM decoder with CoD-based DMD loss.

C. Towards Perceptual Supervision via CoD

In the DMD loss in Equation 1, for any input x, we can optimize its KL divergence using CoD to estimate real and fake scores. This indicates that CoD can act as a perceptual supervision mechanism, enhancing the realism of reconstructions for a variety of networks, including other pixel-space codecs, restoration models, or super-resolution models. To demonstrate this potential, we finetune the pixel-space codec MS-ILLM [11] using the CoD-based DMD loss.

Evaluation. MS-ILLM is typically pretrained using the MSE loss, after which the encoder and entropy model are fixed while the decoder is finetuned with perceptual losses such as LPIPS and GAN [3] loss. Following this paradigm, we finetune only the decoder of MS-ILLM, replacing the GAN loss with the CoD-based DMD loss. Remarkably, after optimizing the decoder for just 2K steps with a batch size of 32 (20K steps total including fake score learning), MS-ILLM demonstrates substantially improved perceptual quality across all metrics, as shown in Figure 5. Towards ultra-low bitrates like 0.011 bpp where the information is severely distorted, we find increasing the learning rate of fake score network training can better capture the distorted distribution to enhance realism, as shown in Table 1. Several visual examples are presented in Figure 6, where the

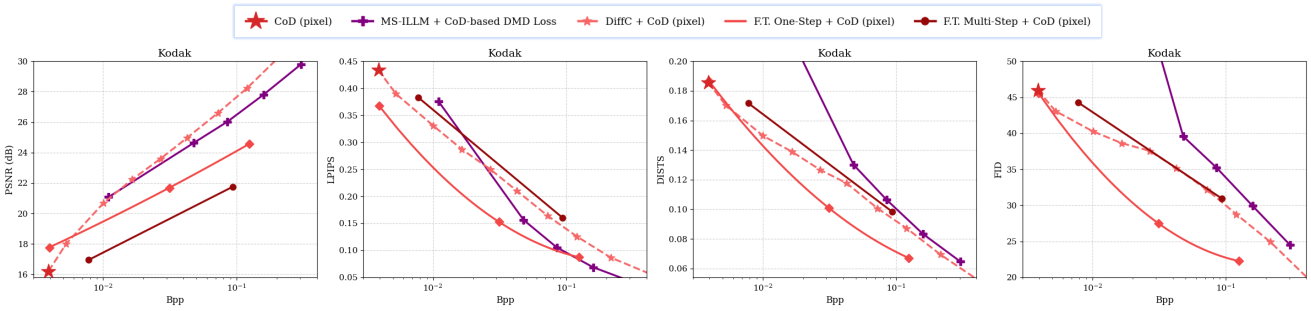

Figure 7. Rate-Distortion curves for different downstream codecs built on pixel-space CoD on Kodak at 512×512.

CoD-based DMD loss markedly reduces artifacts and yields clearer edges and finer details. Since the encoder is fixed, the encoded information remains heavily distorted under MSE-only optimization, resulting in poor overall reconstruction. We expect that jointly tuning the entire network with the CoD-based DMD loss could further improve perceptual quality.

Discussion. Instead of relying on text conditions, CoD learns native image conditions for diffusion. Under DMD supervision, conditions are important since they guide which types of realistic details to generate. Native image conditions provide directions that more closely align with the original image, making CoD theoretically better suited as DMD supervision for reconstruction-related tasks. Moreover, CoD is text-free, eliminating the need for extensive captioning at each optimization step and thereby improving training efficiency. In addition, adopting latent diffusion for DMD loss is non-trivial for pixel-space codecs, whereas our pixel-space CoD offers a more direct and compatible solution.

D. Analysis on Different Downstream Codecs

In this section, we further demonstrate the capability of CoD as a foundation model through comparing different downstream codecs in Figure 7.

D.1. Comparison of Downstream Codecs

Zero-shot DiffC. No perceptual optimization is involved. It yields the highest PSNR and strong perceptual quality, but has slow encoding speed. Runtime grows with bitrate, and encoding a 4 bpp image can take up to 100 seconds.

Finetuned one-step CoD. This scheme attains the best perceptual scores, especially DISTS and FID. The single-step diffusion process enables very fast encoding. However, end-to-end perceptual finetuning with perceptual losses leads to relatively lower PSNR.

CoD-based perceptual optimization for MS-ILLM. Although slightly worse than zero-shot DiffC, it requires a lightweight model and offers substantially faster coding, providing a different balance between coding speed and compression ratio.

D.2. Finetuning Multi-step CoD at Higher Bitrates

A straightforward downstream compression scheme is to directly finetune multi-step CoD at higher bitrates through diffusion loss. However, this leads to inferior results compared to other schemes. When jointly optimizing a condition under explicit diffusion loss, the optimization becomes biased: the condition tends to maximize the score norm, as shown in [21]. In practice, this drives the decoder toward oversharpened and over-saturated reconstructions, resulting in lower fidelity. This observation is consistent with recent findings that one-step diffusion codecs perform better because they avoid this biased optimization.

The tendency toward a higher score norm also motivates training the foundation model at ultra-low bitrates. Under a strong information bottleneck, the condition can only preserve a small amount of information. The distortion is large at these rates, forcing the available condition to focus on reducing diffusion loss rather than amplifying the score norm. At higher bitrates, reducing diffusion loss becomes easier, leaving room for the condition to optimize for score norm instead. In addition, our unified training strategy and auxiliary losses further help counteract this bias by providing additional compression supervision to the condition network.

E. Exploring $\mathcal{X}$-Prediction Pixel Diffusion

Since $\mathcal{V}$-prediction (velocity prediction) was proposed [15] and later adopted by Stable Diffusion v2.1, it has become a common practice in latent diffusion models, including the baseline diffusion architectures used in our work [19, 20]. However, recent research by Just Image Diffusion (JiT) [10] shows that pixels lie on a low-dimensional manifold, making $\mathcal{X}$-prediction a more effective approach for directly modeling pixel distributions. In this paper, we primarily adopt $\mathcal{V}$-prediction to provide a unified view of latent- and pixel-space compression. Nevertheless, in this section, we conduct experiments to demonstrate that $\mathcal{X}$-prediction can achieve better reconstruction performance.

Comparison on pixel-space CoD. We follow the JiT training pipeline to build a pixel-space CoD variant using $\mathcal{X}$-prediction and $\mathcal{V}$-loss, denoted as CoD (pixel-$\mathcal{X}$) to distin-

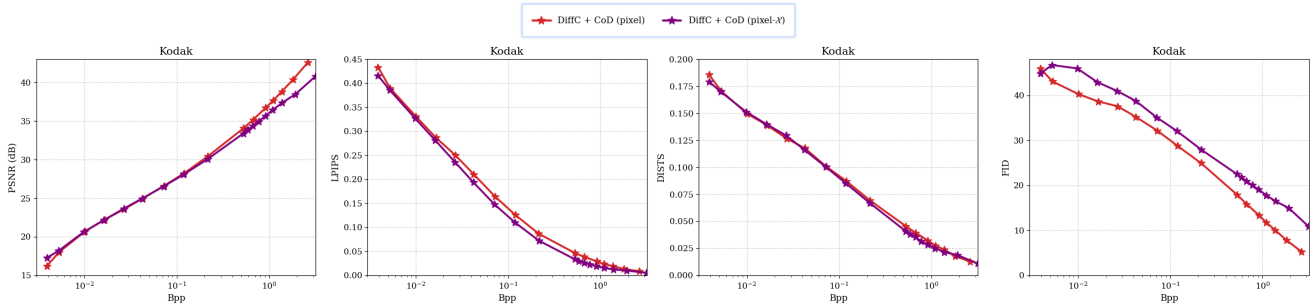

Figure 8. Evaluating $\mathcal{V}$-and $\mathcal{X}$-prediction pixel-space CoD using DiffC on Kodak at 512×512.

Method @ 0.0039 bpp	PSNR	LPIPS	DISTS	FID
CoD (pixel)	16.21 dB	0.434	0.186	46.0
CoD (pixel-$\mathcal{X}$)	**17.29 dB**	**0.416**	**0.179**	**44.8**
CoD (pixel-$\mathcal{X}$, JiT)	16.89 dB	0.422	0.183	47.4

Table 2. Ablation study for $\mathcal{X}$-prediction on Kodak at 512×512.

guish it from our velocity-based pixel-space CoD. To further examine the influence of network design (i.e., the decoupled head [19, 20]) we also substitute our pixel-space diffusion model with JiT's DiT backbone, referred to as CoD (pixel-$\mathcal{X}$, JiT). As shown in Table 2, $\mathcal{X}$-prediction yields significantly better perceptual metrics than $\mathcal{V}$-prediction. In contrast, replacing the decoupled-head design with JiT's pure DiT structure does not provide performance gains.

Evaluation on downstream DiffC. In Figure 8, we evaluate CoD (pixel-$\mathcal{X}$) on the downstream codec DiffC. Although $\mathcal{X}$-prediction improves performance on the foundation model, it does not consistently benefit DiffC. This is because converting the predicted x into velocity v at time t requires clamping the denominator to be no smaller than 0.05 for stability (following JiT): $v = (x - x_t) \,/\, \text{clamp}(1 - t, 0.05, 1)$. This clamping leads to inaccurate likelihood estimation near $t = 1$. When applied to DiffC, the resulting reconstructions retain slight noisy, i.e., the inversion process does not fully denoise the image, leading to degraded FID. The effect is more pronounced at very high bitrates, where the inaccurate likelihood directly impacts DiffC and causes all metrics to drop relative to $\mathcal{V}$-prediction. These results highlight that different prediction targets offer different strengths, and the choice should be adapted to the task objective.

F. Additional Details and Results

This section provides further training details and additional evaluation results, including complexity analysis, additional evaluation metrics and visual comparisons.

F.1. Training Details

F.1.1 Datasets

We train the model using three publicly available datasets: **ImageNet-21K** [14] contains 14.2M images across 21K categories. After removing images with a shorter edge below 256 pixels, 9.3M images remain for 256×256 pre-training. **OpenImages** [8] and **SA-1B** [7] provide high-resolution images. We use 1.7M directly downloadable images from OpenImages V4 and all 11.1M images from SA-1B for both 256×256 and 512×512 resolution pre-training. Overall, CoD is trained on 22.1M images. Compared with modern diffusion pipelines, this is a relatively modest data scale, and the only filtering criterion is image resolution rather than extensive data cleaning. We expect that scaling up training with higher-quality data will further improve the performance of CoD.

F.1.2 Unified Training

Section 2.2 (in the paper) emphasizes that unified training is crucial for achieving high reconstruction fidelity. Table 5 reinforces this observation: unified training (ID = C) provides a clear improvement over the baseline (ID = A). Without unified optimization of the condition and diffusion branches, the condition model tends to encode only structural information while neglecting essential color cues. An intuitive workaround is to impose explicit supervision on the condition learning using an auxiliary loss $\mathcal{L}_{\text{aux}}$.

Auxiliary Loss. Given the output of the condition decoder, we attach two lightweight auxiliary prediction heads to (1) reconstruct the input x and (2) predict its DINO v2 [13] representation. The auxiliary objective is

$$\mathcal{L}_{\text{aux}} = \mathcal{L}_{\text{aux}}^{\text{MSE}} + 0.5 \cdot \mathcal{L}_{\text{aux}}^{\text{DINO}} \tag{3}$$

where $\mathcal{L}\text{aux}^{\text{MSE}}$ measures pixel reconstruction accuracy, and $\mathcal{L}_{\text{aux}}^{\text{DINO}}$ promotes feature consistency by maximizing the cosine similarity between predicted and ground-truth DINO representations. Each auxiliary head consists of a lightweight three-layer convolutional module, adding negligible training overhead. All auxiliary heads are removed at

Method	Stage	Image Resolution	BPP / Total Bits	#Images	Training Steps	Batch Size	Learning Rate	GPU hours (A100)
CoD	Low-Resolution Pre-Training	256×256	0.0156 bpp / 1024 bits	22.1M	400 K	4×32	1×10^{-4}	4×67
CoD	High-Resolution Pre-Training	512×512	0.0039 bpp / 1024 bits	12.8 M	100 K	4×16	2×10^{-5}	4×25
CoD	Unified Post-Training	512×512	0.0039 bpp / 1024 bits	12.8 M	50 K	4×16	2×10^{-5}	4×24

Table 3. Detailed configuration of each CoD training stage. The full training process takes 464 A100 GPU hours (approximately 20 days).

Speed (ms) / Params.	Per-Module Breakdown			Steps	Total
	Conditioner	Diffusion	VAE Decoder		
Stable Diffusion v1.5	203.0 / 3.7 B*	30.6 / 860 M	43.4 / 49 M	25	1011 / 4.6 B
Latent-space CoD	8.3 / 177 M	21.5 / 676 M	43.4 / 49 M	25	589.2 / 901 M
Pixel-space CoD	8.3 / 177 M	25.5 / 720 M	- / -	25	645.8 / 897 M
One-Step Pixel-space CoD	8.3 / 177 M	16.9 / 513 M**	- / -	1	25.2 / 690 M

* Using BLIP2 captioner with at most 32 tokens, as suggested by PerCo (SD) [9].
** The AdaLN-Zero layers are omitted since they become redundant when inference operates with a fixed $t = 0$.

Table 4. Complexity comparison with Stable Diffusion. Average speed (ms) is measured for 512×512 on A100.

ID	Ablation @ 0.0039 bpp	PSNR	LPIPS	FID
A	Flow matching loss	9.83 dB	0.576	76.8
B	A + Auxiliary Loss	15.20 dB	0.458	48.3
C	A + Unified Training	15.83 dB	0.433	48.4
D	B + Unified Post-Training	**16.20 dB**	**0.433**	**46.0**

Table 5. Ablation study on unified training and auxiliary loss on Kodak at 512×512.

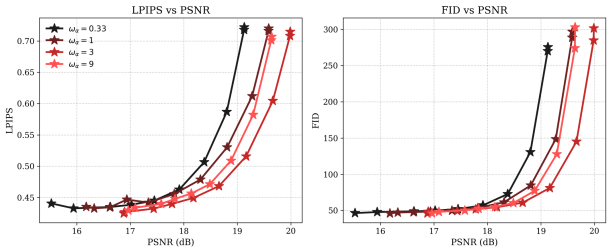

Figure 9. Ablation study on ω_α in unified post-training.

inference time. As shown in Table 5, auxiliary supervision alone (ID = B) achieves FID competitive with unified training, but exhibits lower PSNR and LPIPS since the diffusion model does not receive direct distortion supervision.

Unified Post-Training. Motivated by the complementary strengths of unified training and auxiliary supervision, we combine both approaches to further improve performance. Specifically, we first pre-train CoD with auxiliary loss and then post-train it using unified training. For simplicity, rather than randomly selecting a subset of $\alpha\%$ samples for one-step training, we apply a unified objective to all samples during post-training: $\mathcal{L}_{\text{RF}} = \omega_\alpha \cdot \mathcal{L}_{\text{RF}}^{\text{one-step}} + \mathcal{L}_{\text{RF}}^{\text{multi-step}}$, where ω_α serves as a weighting factor for the one-step loss component. This brings a little better performance. As shown in Table 5, this unified post-training strategy (ID = D) achieves the best overall results among all configurations.

Metric	PerCo	OSCAR	DiffC	DDCM	OneDC	CoD
User Score (1-5) ↑	3.3	2.0	1.8	1.6	3.4	**4.2**
CLIP Sim. ↑	0.84	0.78	0.72	0.63	0.84	**0.89**
Caption Sim. ↑	0.87	0.83	0.76	0.72	0.86	**0.89**
VQA Acc. ↑	69%	66%	58%	48%	71%	**80%**

Table 6. Additional evaluation metrics including user study and semantic scores around 0.004 bpp on Kodak.

We adopt $\omega_\alpha = 1$ as the default setting for unified post-training. In Figure 9, we conduct a sensitivity analysis of this parameter, and the results demonstrate that a value of $\omega_\alpha = 3$ yields additional performance gains, suggesting that further optimization of this hyperparameter could enhance overall efficacy.

F.1.3 Multi-Stage Training

In Table 3, we summarize the configuration of all CoD training stages. A key design choice is to keep the total amount of transmitted information fixed across different resolutions. Concretely, we allocate 0.0156 bpp at 0.0156 bpp at 256×256 and 0.0039 bpp at 512×512, which correspond to the same total bottleneck size of 1024 bits. The codebook size remains unchanged across resolutions. Instead, we change the downsample ratio in the encoder, i.e., $16\times$ at 256×256 and $32\times$ at 512×512.

F.2. Complexity Analysis

Table 4 compares the computational cost of Stable Diffusion v1.5 and CoD. Stable Diffusion requires a large captioning model to generate text conditions, whereas CoD relies on a lightweight 177M image encoder and decoder, taking only 8.3 ms to process a 512×512 image. The diffusion module in CoD also incurs lower latency and parameter overhead. Pixel-space CoD is slightly slower than latent-space CoD

due to the additional decoupled pixel head, but it avoids the expensive VAE decoding, which makes it faster in the few-step regime.

Compared to multi-step diffusion codecs, the one-step pixel-space CoD offers significantly faster inference with a much smaller parameter footprint (see Section B). It enables real-time coding, achieving 25.2 ms per 512×512 image. In the future, we hope it can be further accelerated by training a smaller model. Section C demonstrates that CoD can be used as a perceptual supervisory signal, suggesting a promising path toward fast inference: **train a lightweight CoD and distill it into a single step using the large CoD as perceptual supervision**. We leave this direction to future work with the goal of real-time high-resolution diffusion codecs.

F.3. Additional Evaluation Metrics

Table 6 presents additional performance comparison on the Kodak dataset, with all codecs constrained to approximately 0.004 bpp to ensure a fair evaluation. For the subjective user study, 20 participants rated the reconstruction quality on a scale of 1 (lowest) to 5 (highest), from which Mean Opinion Scores (MOS) were derived. To evaluate semantic preservation, we employed three distinct metrics: (1) CLIP-based visual similarity between original and reconstructed embeddings; (2) text-based similarity between BLIP-2 generated captions of the reconstructions and the ground truth; and (3) a multi-choice Visual Question Answering (VQA) benchmark consisting of 10 GPT-4-designed questions per image. Across all subjective and semantic dimensions, CoD consistently outperforms existing codecs, underscoring its superior generative fidelity.

F.4. Visual Comparison

In Figure 10, we provide more visual comparison examples. Across a wide bitrate range, CoD-based DiffC presents higher perceptual quality than other codecs.

References

[1] Eastman Kodak Company. Kodak lossless true color image suite. `http://r0k.us/graphics/kodak/`, 1999. Accessed: 2025-11-08.

[2] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[4] Jinpei Guo, Yifei Ji, Zheng Chen, Kai Liu, Min Liu, Wang Rao, Wenbo Li, Yong Guo, and Yulun Zhang. Oscar: One-step diffusion codec across multiple bit-rates. *arXiv preprint arXiv:2505.16091*, 2025.

[5] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[6] Zhaoyang Jia, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Generative latent coding for ultra-low bitrate image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26088–26098, 2024.

[7] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[8] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020.

[9] Nikolai Körber, Eduard Kromer, Andreas Siebert, Sascha Hauke, Daniel Mueller-Gritschneder, and Björn W. Schuller. Perco (sd): Open perceptual compression. *CoRR*, 2024.

[10] Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025.

[11] Matthew J Muckley, Alaaeldin El-Nouby, Karen Ullrich, Hervé Jégou, and Jakob Verbeek. Improving statistical fidelity for neural image compression with implicit local likelihood models. In *International Conference on Machine Learning*, pages 25426–25443. PMLR, 2023.

[12] Guy Ohayon, Hila Manor, Tomer Michaeli, and Michael Elad. Compressed image generation with denoising diffusion codebook models. In *Forty-second International Conference on Machine Learning*, 2025.

[13] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

[14] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[15] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

[16] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[17] George Toderici, Lucas Theis, Nick Johnston, Eirikur Agustsson, Fabian Mentzer, Johannes Ballé, Wenzhe Shi, and Radu Timofte. Clic 2020: Challenge on learned image compression. *Retrieved March*, 29:2021, 2020.

[18] Jeremy Vonderfecht and Feng Liu. Lossy compression with pretrained diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[19] Shuai Wang, Ziteng Gao, Chenhui Zhu, Weilin Huang, and Limin Wang. Pixnerd: Pixel neural field diffusion. *arXiv preprint arXiv:2507.23268*, 2025.

[20] Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Ddt: Decoupled diffusion transformer. *arXiv preprint arXiv:2504.05741*, 2025.

[21] Tongda Xu. Optimizing input of denoising score matching is biased towards higher score norm. *arXiv preprint arXiv:2511.11727*, 2025.

[22] Naifu Xue, Zhaoyang Jia, Jiahao Li, Bin Li, Yuan Zhang, and Yan Lu. One-step diffusion-based image compression with semantic distillation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[23] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6613–6623, 2024.

[24] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[25] Tianyu Zhang, Xin Luo, Li Li, and Dong Liu. Stablecodec: Taming one-step diffusion for extreme image compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 17379–17389, 2025.

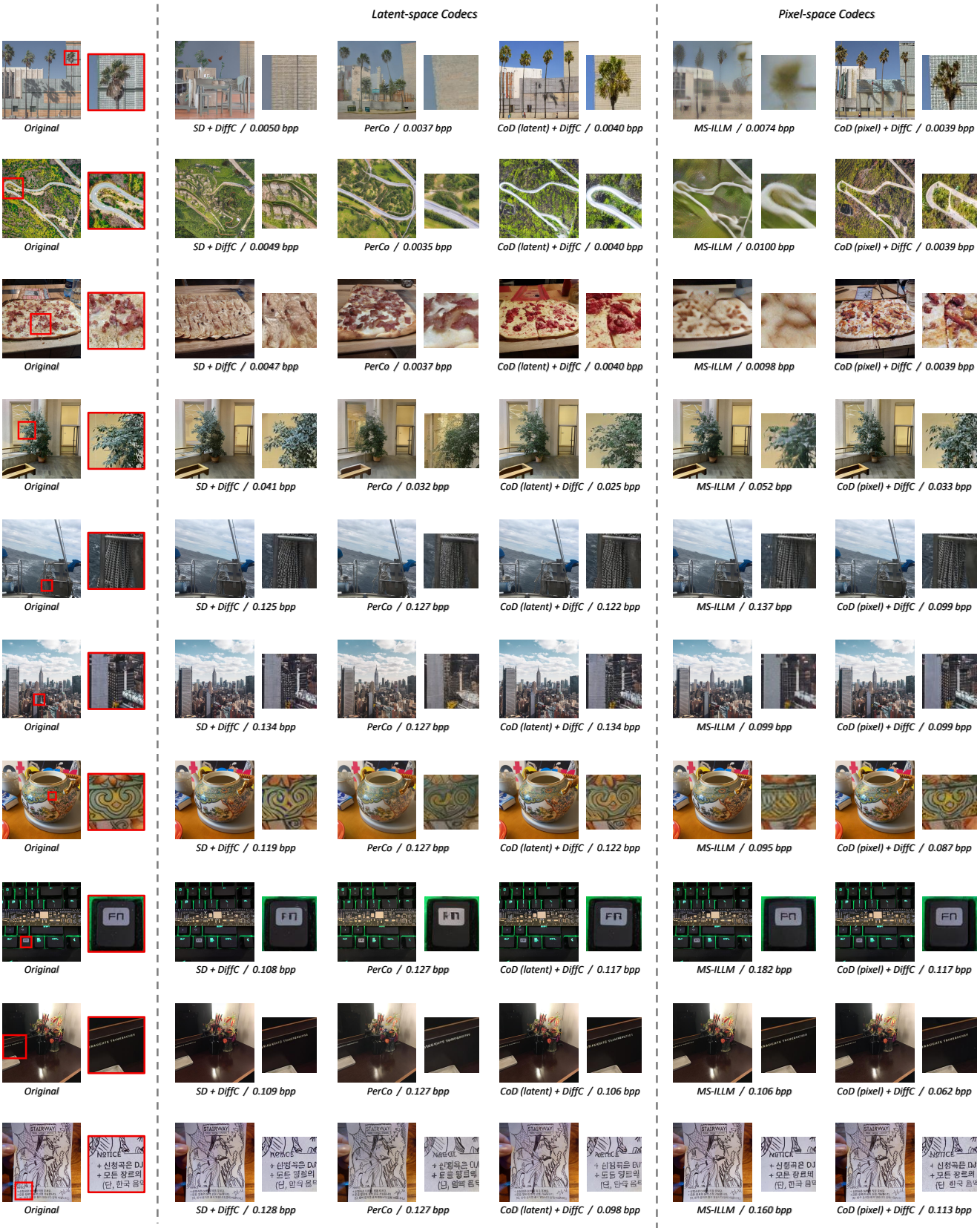

Figure 10. More visualization results on CLIC 2020 test set [17].